Header and macros

Header files contain a stock of library functions for common use. They are named with a `.h` extension. We declare the use of these library functions by including some of these files through the command `#include`. In particular, say if our header file is `header.h`, then `#include<header.h>` looks for the file in the place where it is expected, for example `/usr/include/` whereas `#include "header.h"` looks first for the file in the present working directory. Should that fail it would look next in `/usr/include`.

Example 1 and Programme 1 shows how a header file may be used. The `#ifndef` prevents against multiple inclusion of the same header as multiple declaration may confuse the compiler.

Example 1. (Header file example)

```
1 /* header1.h, a header file, Kit Tyabandha, 23 Jan 07 */
2 #ifndef header1_h
3 #define header1_h
4 #define max(a, b) ((a)>(b)?(a):(b))
5 #endif
```

Programme 1 *Inclusion of the header file in Example 1*

```
/* macros, Kit Tyabandha, 6 Feb 07 */
#include<stdio.h>
#include<stdlib.h>
#include "header1.h"
int
main(){
  int x=4, y=6;
  printf("\n x = %d, y = %d, max(x,y) = %d\n\n", x, y, max(x,y));
  exit(0);
}
```

Header `header1.h` also shows one way to implement a macro. These are no subroutines but expand according to their definition every time they are used in a programme. Following is the compilation and running of Programme 1.

```
kit@nebula:~/prog/c$ make tst
gcc -c -g -Wall tst.c
gcc -g tst.o -o tst -lcurses -ldl -lm
kit@nebula:~/prog/c$ tst
 x = 4, y = 6, max(x,y) = 6
```

Stringising operator

The stringising operator # encloses the argument of #`define` within double quotes. Example 2 gives a header file, `header2.h`, that uses #-operators to turn variable names into strings. Programme 2 uses the macro using this from the header file.

Example 2. (Header file using stringising operators)

```
1 /* header2.h, Kit Tyabandha, 6 Feb 07 */
2 #ifndef header2_h
3 #define header2_h
4 #define max(a, b)\
5    printf("\n "#a" = %.1f\n "#b" = %.1f\n max("#a", "#b") = %.1f\n\n",\
6      (a), (b), ((a)<(b)?(b):(a)))
7 #endif
```

Programme 2 *Use of the header file in Example 2*

```
1 /* stringising operator #, Kit Tyabandha, 6 Feb 07 */
2 #include<stdio.h>
3 #include<stdlib.h>
4 #include "header2.h"
5 int
6 main(){
7   float x=8, y=6;
8   max(x,y);
9   exit(0);
10 }
```

When compiled and run the results were as shown in the following.

```
kit@nebula:~/prog/c$ make tst
gcc -c -g -Wall tst.c
gcc -g tst.o -o tst -lcurses -ldl -lm
kit@nebula:~/prog/c$ tst
 x = 8.0
 y = 6.0
 max(x, y) = 8.0
```